

**spi\_psd\_respgen**

# User Manual

**Version 2.0.1**  
**22 January 2003**

Jürgen Knödseder  
Centre d'Etude Spatiale des Rayonnements  
knodlseder@cesr.fr  
<http://www.cesr.fr/~jurgen/index.html>

#### Note to the user

This software has been written to analyse data of the SPI telescope onboard INTEGRAL. Particular care has been taken in making the software user friendly and well documented. If you appreciated this effort, and if this software and User Manual were useful for your scientific work, the author would appreciate a corresponding acknowledgment in your published work.

## **Contents**

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Getting started</b>	<b>1</b>
<b>3</b>	<b>Parameter file</b>	<b>2</b>
<b>4</b>	<b>Interface definition</b>	<b>4</b>
<b>5</b>	<b>Algorithm</b>	<b>4</b>
<b>6</b>	<b>Error codes</b>	<b>5</b>

## 1 Introduction

The executable `spi_psd_respgen` is part of the SPI scientific analysis preparation software component (SAP). It generates the SPI PSD response function for a given observation group by interpolating PSD response factors that are stored in a `SPI.-DISC-PSD` data structure. Consequently, `spi_psd_respgen` has to be executed after `spi_psd_optimise` but before the imaging or spectral analysis steps. If the response factors `PSD_FB` should be taken into account for background modelling, `spi_psd_respgen` should be executed before `spiback` or `spi_obs_back`. `spi_psd_respgen` adds a data structure of HDU type `SPI.-OBS.-PRF` to the observation group.

`spi_psd_respgen` is written in the ANSI C++ language and has been developed under ISDC support platform 5.1. It requires `spi_toolslib` version 2.0.0 and `spi_psdlib` version 2.0.0 or higher.

## 2 Getting started

Before installing `spi_psd_respgen`, make sure that the ISDC support platform 5.1 or higher is installed on your system, and that you have installed the library `spi_toolslib` version 2.0.0 and `spi_psdlib` version 2.0.0 or higher.

After downloading the `spi_psd_respgen.tar.gz` file, step into a directory that should hold the distribution, move the `spi_psd_respgen.tar.gz` file into this directory and type after the UNIX prompt `$` (don't type this prompt):

```
$ gunzip spi_psd_respgen.tar.gz
$ tar xvf spi_psd_respgen.tar
```

The first command uncompresses the distribution file, the second unpacks the files.

Before configuration, the distribution needs to be reset to a clean state. To do this, type

```
$ make distclean
```

Then, configure the distribution. It is assumed here that you have previously installed the ISDC support platform, thus you should type

```
$ ~/bin/ac_stuff/configure
```

Finally, build the distribution by typing

```
$ make global_install
```

To perform a unit test, type

```
$ make test
```

Make sure that the test data `spi_test_data-1.0.tar.gz` are available at your site (they should reside in a directory whose name is defined by the `ISDC_TEST_DATA_DIR` environment variable).

### 3 Parameter file

```
#####
#
#           Centre d'Etude Spatiale des Rayonnements
#           (in collaboration with ISDC)
#
#           SPI PSD response generation
#
# -----
#
# File:      spi_psd_respgen.par
# Version:   2.0.1
# Component: SAP
#
# Author:    Juergen Knoedlseder
#            knodlseder@cesr.fr
#            CESR
#
# Purpose:   Parameter file of the SPI PSD response generation task
#
# History:   2.0.1 22-Jan-2003 Second ISDC delivery
#
#####
#
# The input DOLs/filenames
#=====
ingrpDOL,s,ql, "og_spi.fits[GROUPING]",,, "Input Observation Group DOL or filename"
inebdDOL,s,ql, "boundaries.fits[SPI.-EBDS-SET]",,, "Energy boundary definition DOL or filename"
ingtiDOL,s,ql, "gti.fits[SPI.-OBS.-GTI]",,, "Good Time Interval DOL or filename"
indisDOL,s,ql, "psd_disc.fits[SPI.-DISC-PSD]",,, "PSD discrimination DOL or filename"
#
# The output DOLs/filenames
#=====
outgrpDOL,s,ql, "og_spi_psd.fits[GROUPING]",,, "Output Observation Group DOL or filename"
outprfDOL,s,ql, "psd_response.fits[SPI.-OBS.-PRF]",,, "PSD response DOL or filename"
#
# Standard parameters
#=====
clobber,b,h, yes,,, "Overwrite existing output data ?"
verbose,i,h, 2,0,3, "Information logging level"
```

Instead of specifying complete DOLs (Data Object Locations), which are composed of a filename plus the data structure extension (HDU), `spi_psd_respgen` accepts also simple filenames and adds the appropriate data structure extensions. This means that **specified data structure extensions are ignored**.

The parameters have the following meaning:

- **ingrpDOL** (optional) specifies the DOL or filename of the input Observation Group of HDU type `[GROUPING]` for which the SPI PSD response function should be derived. The specification of this parameter is optional **if an output Observation Group has been specified** (parameter `outgrpDOL`). If left blank, the output group will be used as input group. This option may be used if one wants to add-on to an existing Observation Group (in this case the `clobber` parameter has to be set to **yes**).

For safety, however, it is recommended to create a new output Observation Group (which will take only little space on your disk). Only an Observation Group is accepted for this parameter, other data or group types (such as science window groups or index groups) will be rejected.

- **inebdDOL** (optional) specifies the DOL or filename of an energy boundary definition file of HDU type `[SPI.-EBDS-SET]`. If a `[SPI.-EBDS-SET]` element exists already in the input Observation Group, this element will be replaced by the specified DOL in the output Observation Group. If no `[SPI.-EBDS-SET]` element exists in the input Observation Group, the specified DOL will be attached to the output Observation Group. Leaving this parameter blank will lead `spi_psd_respgen` to search for this element in the input Observation Group. This is the recommended default method. **Note that `spi_psd_respgen` only works for calibrated energies, hence the energy boundary definition must be specified for calibrated energies (`CHANTYPE=PI`).**
- **ingtiDOL** (optional) specifies the DOL or filename of a Good Time Interval list of HDU type `[SPI.-OBS.-GTI]`. If a `[SPI.-OBS.-GTI]` element exists already in the input Observation Group, this element will be replaced by the specified DOL in the output Observation Group. If no `[SPI.-OBS.-GTI]` element exists in the input Observation Group, the specified DOL will be attached to the output Observation Group. Leaving this parameter blank will lead `spi_psd_respgen` to search for this element in the input Observation Group. This is the recommended default method.
- **indisDOL** (optional) specifies the DOL or filename of a PSD discrimination and response parameter file of HDU type `[SPI.-DISC-PSD]`. If a `[SPI.-DISC-PSD]` element exists already in the input Observation Group, this element will be replaced by the specified DOL in the output Observation Group. If no `[SPI.-DISC-PSD]` element exists in the input Observation Group, the specified DOL will be attached to the output Observation Group. Leaving this parameter blank will lead `spi_psd_respgen` to search for this element in the input Observation Group. This is the recommended default method.
- **outgrpDOL** (optional) specifies the DOL or filename of the output Observation Group of HDU type `[GROUPING]`. The output Observation Group will be a copy of the input Observation Group plus the resulting SPI PSD response function data structure `[SPI.-OBS.-PRF]` attached. The specification of this parameter is optional **if an input Observation Group has been specified** (parameter `ingrpDOL`). If left blank, the input group will be used as output group. This option may be used if one wants to add-on to an existing Observation Group (in this case the `clobber` parameter has to be set to **yes**). For safety, however, it is recommended to create a new output Observation Group (which will take only little space on your disk). Only an Observation Group is accepted for this parameter, other data or group types (such as science window groups or index groups) will be rejected.
- **outprfDOL** specifies the DOL or filename of the SPI PSD response function of HDU type `[SPI.-OBS.-PRF]`. This file will be attached to the output Observation Group. Any existing DOL of the same type that may already exist in the output Observation Group will be detached before. If the specified DOL is identical to an already existing DOL, this DOL will be deleted if the `clobber` parameter is **yes** (otherwise an error will occur).
- **clobber** specifies if existing output data structures should be overwritten or not. If **yes** is specified, the executable will notify the user about the deletion of any file. If **no** is specified and the executable attempts to overwrite existing data (e.g. an existing output Observation Group or PSD response structure), the task will exit with an error message.
- **verbose** specifies the verbose level of the executable. For **verbose=0**, no information will be logged in case of an error. For **verbose=1**, only errors will be logged, while for **verbose=2** also actions (such as DOL detachments and attachments and DOL deletion) will be logged. **verbose=3** provides a detailed report about the SPI PSD response function generation.

## 4 Interface definition

**spi-psd-respgen** takes the information that is available in the PSD discrimination and response parameter data structure (HDU extension **SPI.-DISC-PSD**) and derives from it the PSD response function for a given Observation Group. In addition to this HDU, it also requires a **SPI.-EBDS-SET** and a **SPI.-OBS.-GTI** HDU to determine the energy boundary definition of the event spectra, and the Good Time Intervals and pointings that are present in the Observation Group.

**spi-psd-respgen** stores the calculated SPI PSD response function in a data structure of HDU type **SPI.-OBS.-PRF** where it creates two columns **PSD\_FP** and **PSD\_FB** with the same number of column bins than the event spectra have (i.e. the column size is set by the executable). For each pointing, **SPI.-OBS.-PRF** contains 19 rows, one for each SPI detector (with detector identifiers in the order 0 - 18).

From the input Observation Group, the following HDUs are required:

- **SPI.-EBDS-SET** Energy boundary definition of the observation
- **SPI.-OBS.-GTI** Good Time Intervals
- **SPI.-DISC-PSD** PSD discrimination parameters and response factors

All members of the input Observation Group become members of the output Observation Group. In addition, the following additional HDU is attached to the output Observation Group (or replaced if it exists already):

- **SPI.-OBS.-PRF** PSD response function

If no output Observation Group is specified, **spi-psd-respgen** attempts to make the input Observation Group to the output Observation Group, which is only possible if the **clobber** parameter is set to **yes**. Vice versa, if no input Observation Group is specified, **spi-psd-respgen** uses the output Observation Group as input Observation Group. For more details on the group logic, the user is referred to the discription of the group API in the **spi-toolslib** User Manual.

## 5 Algorithm

The SPI PSD response function consists of the two parameters **PSD\_FP** and **PSD\_FB**.

**PSD\_FP** is the probability that the PSD correctly recognises a photon event as a photon event (i.e. the probability that a photon event that was analysed by the PSD sub-assembly got a **PSD\_FLAG=1**). It is emphasised that only the fraction of analysed events are taken into account, hence events that had an analysis error are not considered (the fraction of events that either escaped the SPI PSD sub-assembly due to deadtime, or that were assigned an analysis error is stored in the column **PSD\_EFF** of the data structure **SPI.-OBS.-PEF**).

**PSD\_FP** is in general detector, energy and time dependent. For a given set of energies and time intervals, the executable **spi-psd-optimize** determines the **PSD\_FP** factors by analysing the SPI PSD discrimination of instrumental background photons for each detector. **spi-psd-optimize** may divide an Observation Group in time intervals of constant **ONTIME**, and therefore may derive time dependent **PSD\_FP** factors. These factors are stored together with the PSD discrimination parameters in a **SPI.-DISC-PSD** data structure.

**spi-psd-respgen** takes these factors and performs a linear interpolation in energy to obtain **PSD\_FP** for each energy bin of the Observation Group and each detector. In order to handle a time dependent PSD response, it determines for each pointing the set of **PSD\_FP** from the **SPI.-DISC-PSD** data structure that has the longest time overlap with the pointing time interval. The pointing time interval is defined by the

earliest and latest time for a given detector and pointing. If for any reason no valid pointing time interval could be determined, the first set of **PSD\_FP** factors that is found in **SPI.-DISC-PSD** is used.

**PSD\_FB** is the probability that the PSD correctly recognises a background event as a background event (i.e. the probability that a background event that was analysed by the PSD sub-assembly got a **PSD\_FLAG=0**). Also these factors are determined by **spi-psd-optimize**, and interpolated and assigned in the same manner by **spi-psd-respgen** as it is the case for **PSD\_FP**. Note that **PSD\_FB** are not directly required by imaging or spectral analysis, yet they may be useful for background modelling.

## 6 Error codes

The executable **spi-psd-respgen** may stop with the following error codes:

<b>SPI_PSD_RESPGEN_ERROR_MEM_ALLOC</b>	<b>-231900</b>
<b>SPI_PSD_RESPGEN_ERROR_OVERWRITE</b>	<b>-231901</b>
<b>SPI_PSD_RESPGEN_ERROR_NOT_PI</b>	<b>-231902</b>

They have the following meaning:

- **SPI\_PSD\_RESPGEN\_ERROR\_MEM\_ALLOC** : the allocation of dynamical memory has failed. Probable your system resources are too limited to run this task.
- **SPI\_PSD\_RESPGEN\_ERROR\_OVERWRITE** : the task attempted to overwrite a data structure while the **clobber** parameter was set to **no**.
- **SPI\_PSD\_RESPGEN\_ERROR\_NOT\_PI** : the **SPI.-EBDS-SET** data structure is not defined for calibrated energies. Yet, **spi-psd-respgen** only can handle calibrated energies since they are needed for response interpolation.

In addition, all errors that may occur in calls to ISDC support functions (such as for example DAL, RIL or PIL) are forwarded. Please consult the ISDC web pages for getting information about these error codes.